

---

Using  
hrPINGv3.12



Tony Fortunato  
Sr Network Performance Specialist  
The Technology Firm

# What is hrPING?



- Go get it at [http://www.cfos.de/ping/ping\\_e.htm](http://www.cfos.de/ping/ping_e.htm)
- Portable, command line utility
  - no installation
  - Make sure you are in admin mode so hrping can create a raw socket \*
- High-precision ping utility with advanced functionality and improved statistics

```
Administrator: C:\Windows\system32\cmd.exe

C:\Users\Tony Fortunato\Documents\ttf\software\hrping-v312>hrping 172.16.2.33
This is hrPING v3.12 by cFos Software GmbH -- http://www.cfos.de

Source address is 10.44.10.107; using ICMP echo-request
Pinging 172.16.2.33
with 32 bytes data (60 bytes IP):

Reply from 172.16.2.33: seq=0001 time=142.664ms TTL=63 ID=8ac0
Reply from 172.16.2.33: seq=0002 time=21.429ms TTL=63 ID=8ac1
Reply from 172.16.2.33: seq=0003 time=15.914ms TTL=63 ID=8ac2
Reply from 172.16.2.33: seq=0004 time=13.661ms TTL=63 ID=8ac3

Statistics for 172.16.2.33:
  Packets: sent=4, rcvd=4, error=0, lost=0 (0.0% loss) in 1.513653 sec
  RTTs of replies in ms: min/avg/max/dev: 13.661 / 48.417 / 142.664 / 54.486
  Bandwidth in kb/sec: sent=0.158, rcvd=0.158

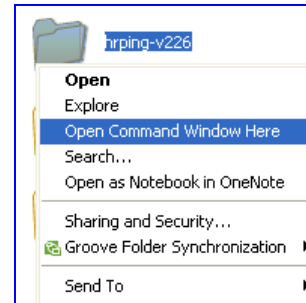
C:\Users\Tony Fortunato\Documents\ttf\software\hrping-v312>
```

\* If you want hrping or other apps access to raw sockets, you can set the following Registry key  
HKEY\_LOCAL\_MACHINE to 1 (DWORD):  
System\CurrentControlSet\Services\Afd\Parameters\DisableRawSecurity



- hrPING can be used for the following tasks;
- Provides more detailed statistics than other PING tools
  - ✓ Sequence Number
  - ✓ Time reference
- Measures the time (in milliseconds) it takes for a packet to travel from your computer to a specified destination, with the accuracy of three decimal places.
- Change the time interval that hrPING sends out a PING packet, while listening for the reply
- Perform a traceroute to determine the distance and number of hops to a specified address.
- Output can be logged and imported into Excel or other graphing tools
  - ✓ Like other PING tools, hrPING sends ICMP "Echo Request" packets to the remote computer and listens to the matching "Echo response" packets
  - ✓ Unlike other PING tools hrping can be used to send UDP packets as well

- To start the hrPING utility you must open the folder in the command prompt (in admin mode).
- If you have the Windows Powertoy, simply right click on the hrPING folder and then open in command prompt.
- Once the command prompt opens into the hrPING folder, type hrping on the command line to review the options.
- Please note that you should go to the command prompt in administrative mode to avoid any issues running hrping. Or modify your registry as per the instructions at the bottom of the previous slide.



```
ex C:\WINDOWS\system32\cmd.exe

C:\Documents and Settings\Tony Fortunato\Desktop\peter\hrPING>hrping
This is hrPING v2.39 by cFos Software GmbH -- http://www.cfos.de

usage: hrPING [options] host

options:
-l lic          Show public license and warranty
-t            Ping the specified host until stopped
-n count      Number of echo requests to send (default 4)
-E file       Stop pinging when <file> exists
-l size       Send buffer size (ICMP payload size, default 64)
-L size       Total IP datagram size (ICMP payload size + 28)
-f           Set Don't Fragment bit in IP header
-i TTL        Time To Live (default 255 for ping, 30 for traceroute)
-v TOS        Type Of Service (default 0)
-w timeout    Timeout in millisecs to wait for each reply (default 2000)
-s time       Interval in millisecs between packets (default 500 ms)
-r [count]    Be a traceroute (do <count> pings each hop, default 3)
-a [hop]      Resolve addresses to names for traceroute (start at <hop>)
-o           Don't do overlapped send/receive
-tsc         Force RDTSC usage
-w           "warm up" with one uncounted echo request at beginning
-F file       Log output into <file> as well
-T           Print timestamp in front of each line
-I id         Set ICMP id field to <id>
-q           Don't print a line per ping
-A           Abort after the first echo reply (-AA => or error)
-H           use IP_HDR_INCL socket option (experimental)
-S           print a summary on each receive

Thank you for using hrPING! hrPING is Freeware, share it with anybody.
Check out www.cfos.de for our Traffic Shaping driver cFosSpeed, improved IPv6
Connectivity with cFos IPv6 Link or cFos Broadband Connect for faster PPPoE!
```

# hrPING Options (from the command line usage text)



Data Options	[value] indicates this value is optional
<b>-f</b>	Set Don't Fragment flag in packet. Set the "Don't fragment" bit in the IP header of the PING packet. Default is not set.
<b>-i TTL</b>	Time To Live (default 255 for ping, 30 for traceroute)
<b>-v TOS</b>	Type Of Service (default 0)
<b>-l size</b>	Send buffer size (payload size, default 32)
<b>-L size</b>	Total IP datagram size (payload size + 28, default 60)
<b>-I id</b>	Set ICMP id field to <id>
<b>-M</b>	Send ICMP timestamp requests (-MM => more output)
<b>-u [port]</b>	Send UDP packets (port 7 by default)



Operational Options	
<b>-t</b>	Ping the specified host until stopped. Loop forever. You can abort <i>hrPING</i> any time with CTRL-C or CTRL-Break. Unlike Windows PING, <i>hrPING</i> will still print the statistics gathered so far when you abort. CTRL-C waits for some time for replies still to come in, while CTRL-Break aborts right away.
<b>-n count</b>	Number of echo requests to send. Specify the number of PING packets to send. Default number is 4.
<b>-E file</b>	Stop pinging when <file> exists This is nice for batch files or for coordinating with a background job. <i>hrPING</i> will loop as long as usual (i.e. depending on -t or -n options), but will furthermore check for the existence of <file>. If <file> comes into existence, <i>hrPING</i> will exit the loop.
<b>-w timeout</b>	Timeout in milliseconds to wait for each reply. Maximum timeout to wait for a reply. This is almost only of use if you switch to non-overlapped (-o) mode. In overlapped mode, this time only applies when <i>hrPING</i> has stopped sending (because the count was exceeded or because you pressed CTRL-C) and is waiting for missing replies. Default is 2000 milliseconds.
<b>-s time</b>	Interval in milliseconds between packets. This is the number of milliseconds between sending of two PING packets. <i>hrPING</i> will try to stick to this number very accurately. If sending took a little longer for one packet it will send out the next packet a little earlier. Default is 500 milliseconds. (You can use decimals for a very fine grained interval: -s5.4 will send a packet every 5400 microseconds, on average!)
<b>-r [count]</b>	Switch to traceroute mode. <i>hrPING</i> contains a traceroute utility! It works almost the same as Windows TRACERT, except that it only does one test per host, not three. By default, IP addresses are not resolved to names. Use -a to do that. <b>Really does 3 pings per TTL.</b>
<b>-a [hop]</b>	Resolve addresses to hostnames in traceroute mode. No need to say more.

Operational Options	
<b>-o</b>	Don't do overlapped send/receive. Use Windows PING like synchronous sending of one packet, waiting for the reply and so on. Off by default.
<b>-tsc</b>	Force RDTSC usage. hrPING automatically decides if it uses the CPU's timestamp counter (TSC) or the operating system's performance counter for timings. On some CPU's the TSC is not reliable, since it doesn't tick at the same speed all the time. On multiprocessor systems, not all TSC have to tick exactly in sync. In almost all cases, hrPING will use the performance counter. If you want to force TSC usage, use <code>-tsc</code> .
<b>-W</b>	"warm up" with one uncounted echo request at beginning If specified, hrPING will send one uncounted ping before all others. This "warm up" is useful with some firewalls that somehow cause the first block to be much slower than the following ones.
<b>-A</b>	Abort after the first echo reply (-AA => or error) Loop as long as there are no replies (or even error messages if -AA).
<b>-H</b>	use IP_HDRINCL socket option to send packets

Output Options	
<b>-lic</b>	Show public license and warranty
<b>-fwhelp</b>	Print firewall help text
<b>-F file</b>	Log output into <file> as well
<b>-T</b>	Print timestamp in front of each line. Preceeds each line of output with a timestamp of the form "2006-11-22 10:55:27.201: "
<b>-q</b>	Don't print a line per ping Be quiet.
<b>-S</b>	Print a summary on each receive
<b>-O ofs</b>	Set time offset in msec for timestamp mode
<b>-y [sec]</b>	Print summary (for the last <sec> secs), not one line each ping



# hrPING vs Windows PING

---

Description	hrPing	Windows
Continuous ping	-t	-t
Name resolution	-a	-a
# of requests to send	-n count	-n count
Send buffer size	-l size	-l size
Set don't fragment bit	-f	-f
Time to live	-i TTL	-i TTL
Type of service	-v TOS	-v TOS
Do traceroute	-r [count]	NA
IP timestamp	NA	-s count
Timeout to wait for reply	-w timeout	-w timeout
Don't overlap send/receive	-o	NA
Force RDTSC usage	-tsc	NA
Warm up with request	-W	NA
Log output to <file>	-F file	redirect
Print timestamp	-T	NA
Set ICMP id field to <id>	-l id	NA
Don't print a line per ping	-q	NA

# hrPING vs Windows PING

---

- As you can see in the example below, both hrPING and the Windows PING utilities look very similar.
- What you should notice is that hrPING shows you a sequence number, a more granular time value, and an ID number..
- hrPING also gives the percentage of errors that occurred along with the sent, received, and lost.

*Windows PING*

```
>ping 172.16.4.1
Pinging 172.16.4.1 with 32 bytes of data:
Reply from 172.16.4.1: bytes=32 time=4ms TTL=254
Reply from 172.16.4.1: bytes=32 time=3ms TTL=254
Reply from 172.16.4.1: bytes=32 time=3ms TTL=254
Reply from 172.16.4.1: bytes=32 time=3ms TTL=254

Ping statistics for 172.16.4.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 3ms, Maximum = 4ms, Average = 3ms
```

*hrPING*

```
>hrping 172.16.4.1
This is hrPING v3.12 by cFos Software GmbH -- http://www.cfos.de
Source address is 10.44.10.107; using ICMP echo-request
Pinging 172.16.4.1
with 32 bytes data (60 bytes IP):

Reply from 172.16.4.1: seq=0001 time=5.134ms TTL=254 ID=60b1
Reply from 172.16.4.1: seq=0002 time=4.235ms TTL=254 ID=60b2
Reply from 172.16.4.1: seq=0003 time=3.563ms TTL=254 ID=60b3
Reply from 172.16.4.1: seq=0004 time=3.587ms TTL=254 ID=60b4

Statistics for 172.16.4.1:
    Packets: sent=4, rcvd=4, error=0, lost=0 (0.0% loss) in 1.503577 sec
    RTTs of replies in ms: min/avg/max/dev: 3.563 / 4.129 / 5.134 / 0.639
    Bandwidth in kb/sec: sent=0.159, rcvd=0.159
```

# Getting The Results

---

## Statistics for 172.16.4.1:

**Packets: sent=4, rcvd=4, error=0, lost=0 (0.0% loss) in 1.503697 sec**

- sent = how many pings you sent
- rcvd = how many pings you received
- error = how many icmp errors you received
- lost = how many packets you did not receive (within your timeout -w value)

**RTTs of replies in ms: min/avg/max/dev: 3.459 / 3.684 / 3.848 / 0.145**

- Min = fastest response time
- Avg = average response time
- Max = slowest response time
- Dev = standard deviation (a lower value illustrates that the data is tighter together, or in most cases better)

**Bandwidth in kb/sec: sent=0.159, rcvd=0.159**

- This records the amount of bandwidth consumed during your test

# Interpreting The Results

---

**Statistics for 172.16.4.1:**

**Packets: sent=4, rcvd=4, error=0, lost=0 (0.0% loss) in 1.503697 sec**

- sent = how many pings you sent
  - rcvd = how many pings you received
  - error = how many icmp errors you received
  - lost = how many packets you did not receive (within your timeout -w value)
- 
- Sent and received are straight forward. Lets look at error and lost in more detail.

# Interpreting The Results - Lost

---

- To better understand timeouts, we need a protocol analyzer (like Wireshark) to validate what we see
- Firstly, we will perform a simple *hrping 172.16.2.33* and capture the packets with Wireshark.
  - ✓ Within Wireshark's capture option -> capture filter, we type in **icmp** so we only capture those packets

```
>hrping 172.16.2.33
This is hrPING v3.12 by cFos Software GmbH -- http://www.cfos.de

Source address is 10.44.10.107; using ICMP echo-request
Pinging 172.16.2.33
with 32 bytes data (60 bytes IP):

Reply from 172.16.2.33: seq=0001 time=71.645ms TTL=63 ID=8b77
Reply from 172.16.2.33: seq=0002 time=20.378ms TTL=63 ID=8b78
Reply from 172.16.2.33: seq=0003 time=18.677ms TTL=63 ID=8b79
Reply from 172.16.2.33: seq=0004 time=53.449ms TTL=63 ID=8b7a

Statistics for 172.16.2.33:
  Packets: sent=4, rcvd=4, error=0, lost=0 (0.0% loss) in 1.553437 sec
  RTTs of replies in ms: min/avg/max/dev: 18.677 / 41.037 / 71.645 / 22.459
  Bandwidth in kb/sec: sent=0.154, rcvd=0.154
```

- Now we know our response time is between 18 and 71 ms, so lets try the same hrping with the **-w 10** timeout option and we will set it to 10 ms
- Now we expect all pings with a higher response time than 10 ms to fail

# Interpreting The Results - Lost

---

```
>hrping 172.16.2.33 -w 10
This is hrPING v3.12 by cFos Software GmbH -- http://www.cfos.de

Source address is 10.44.10.107; using ICMP echo-request
Pinging 172.16.2.33
with 32 bytes data (60 bytes IP):

Reply from 172.16.2.33: seq=0001 time=45.627ms TTL=63 ID=8b89
Reply from 172.16.2.33: seq=0002 time=19.684ms TTL=63 ID=8b8a
Reply from 172.16.2.33: seq=0003 time=21.612ms TTL=63 ID=8b8b
1 request timed out.

Statistics for 172.16.2.33:
  Packets: sent=4, rcvd=3, error=0, lost=1 (25.0% loss) in 1.021570 sec
  RTTs of replies in ms: min/avg/max/dev: 19.684 / 28.974 / 45.627 / 11.801
  Bandwidth in kb/sec: sent=0.234, rcvd=0.176
```

- What the !\$#!?! ? All 4 pings should have timed out.
- Here's the funny part, most people I show this to dismiss it as, "what do you expect from freeware".
- I counter with, lets reread what the `-w` option states in my slide notes;
  - ✓ "this time only applies when *hrPING* has stopped sending"
- In this case, it applies to the last ping
- So lets test our theory

# Interpreting The Results - Lost

---

- Lets start with a single ping to a host `ping 172.16.2.33 n 1`

```
Reply from 172.16.2.33: seq=0001 time=49.669ms TTL=63 ID=8bbd
```

- Great, lets do the same ping with a `-w 10`

```
1 request timed out.
```

- Interesting, and equally good to know
- Moral of the story, `-w` works best with singular pings `-n`
- If you do send multiple pings, then pay attention to the last one
- `-w` works best **AFTER** you determine your current response times
  
- Lets move on to **ERROR**

# Interpreting The Results - Error

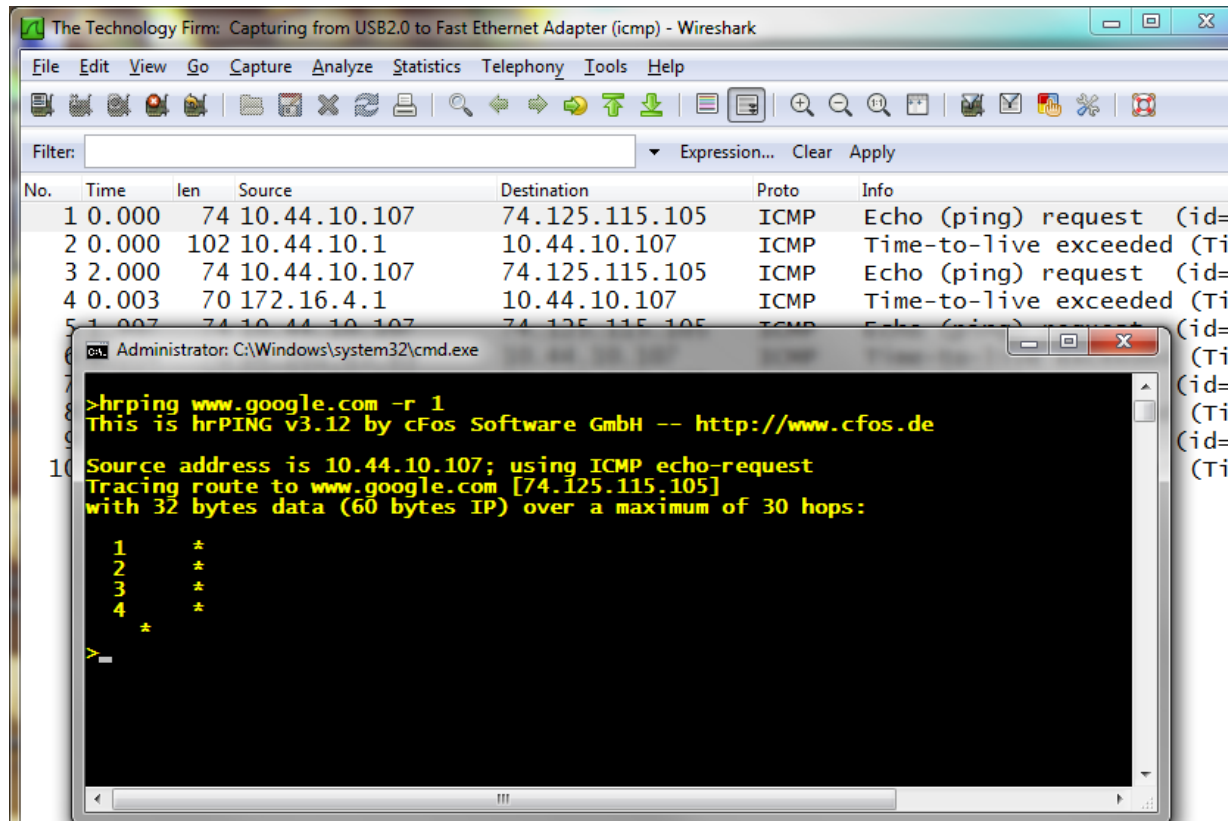
---

- What are some examples of ICMP Errors;
  - ✓ Destination Unreachable
    - ❖ This applies when an IP host (like a router) sends back an ICMP packet, not a packet that simply times out
  - ✓ Time Exceeded
    - ❖ A packet's IP Time to live value was been decremented to 0 and a router or firewall sent this back
  - ✓ Redirect Messages
    - ❖ A router or Firewall sent informed you of a better ip address to use for your destination



# Interpreting The Results – Error – The Test

- Lets try to replicate the Time To Live Error with one test per route using *hrping* [www.google.com](http://www.google.com) -r 1



- If I look at my trace, I can see that I am getting ICMP TIME TO LIVE EXCEEDED, but hrping doesn't record it
- ... Wait a minute, I think I know what this is

# Interpreting The Results – Error – The Test

- I will check my firewall settings;

## Customize settings for each type of network

You can modify the firewall settings for each type of network location that you use.

What are network locations?

### Home or work (private) network location settings



Turn on Windows Firewall

Block all incoming connections, including those in the list of allowed programs

Notify me when Windows Firewall blocks a new program



Turn off Windows Firewall (not recommended)

### Public network location settings



Turn on Windows Firewall

Block all incoming connections, including those in the list of allowed programs

Notify me when Windows Firewall blocks a new program



Turn off Windows Firewall (not recommended)

- I will try this again, with my firewall disabled

- Yippee, it worked..

- I normally have my firewall disabled in the lab, but forgot I enabled my firewall when I was presenting

```
>hrping www.google.com -r 1
This is hrPING v3.12 by cFos Softwa

Source address is 10.44.10.107; usi
Tracing route to www.google.com [74
with 32 bytes data (60 bytes IP) ov

 1    0.820    [10.44.10.1]
 2    3.433    [172.16.4.1]
 3    5.511    [64.201.63.89]
 4    5.495    [216.185.64.25]
 5    5.558    [216.185.67.171]
 6    6.966    [38.112.1.49]
 7   18.766    [154.54.40.177]
 8   18.458    [154.54.42.73]
 9   18.952    [154.54.5.210]
10   19.389    [4.68.111.45]
```

# Testing the TTL or -i

- Now that everything is working, lets try to hrping with a TTL that is intentionally too low
- I will use the results from the previous test and choose the ip at hop 6 and hrping it with a ttl of 4 **hrping 38.112.1.49 -i 4**
- I use this trick when I want to see if an intermediate router is flapping or having performance issues

```
15:57:24.44> hrping 38.112.1.49 -i 4
This is hrPING v3.12 by cFos Software GmbH -- http://www.cfos.de

Source address is 10.44.10.103; using ICMP echo-request
Pinging 38.112.1.49
with 32 bytes data (60 bytes IP), TTL 4:

Reply from 66.207.97.25: TTL count exceeded; seq=0001 time=6.575ms TTL=252 ID=0000
Reply from 66.207.97.25: TTL count exceeded; seq=0002 time=5.585ms TTL=252 ID=0000
Reply from 66.207.97.25: TTL count exceeded; seq=0003 time=5.581ms TTL=252 ID=0000
Reply from 66.207.97.25: TTL count exceeded; seq=0004 time=5.479ms TTL=252 ID=0000

Statistics for 38.112.1.49:
Packets: sent=4, rcvd=0, error=4, lost=0 (0.0% loss) in 1.505475 sec
RTTs of errors in ms: min/avg/max/dev: 5.479 / 5.805 / 6.575 / 0.446
Bandwidth in kb/sec: sent=0.159, rcvd=0.148
```

No.	Time	len	Source	Destination	Proto	Info
1	0.000	74	10.44.10.103	38.112.1.49	ICMP	Echo (ping) request id=0x9c10, seq=1/256, ttl=4
2	0.006	70	66.207.97.25	10.44.10.103	ICMP	Time-to-live exceeded (Time to live exceeded in transit)
3	0.493	74	10.44.10.103	38.112.1.49	ICMP	Echo (ping) request id=0x9c10, seq=2/512, ttl=4
4	0.005	70	66.207.97.25	10.44.10.103	ICMP	Time-to-live exceeded (Time to live exceeded in transit)
5	0.494	74	10.44.10.103	38.112.1.49	ICMP	Echo (ping) request id=0x9c10, seq=3/768, ttl=4
6	0.005	70	66.207.97.25	10.44.10.103	ICMP	Time-to-live exceeded (Time to live exceeded in transit)
7	0.494	74	10.44.10.103	38.112.1.49	ICMP	Echo (ping) request id=0x9c10, seq=4/1024, ttl=4
8	0.005	70	66.207.97.25	10.44.10.103	ICMP	Time-to-live exceeded (Time to live exceeded in transit)

---

Interpeting  
hrPINGv3.12  
Results



Thank You For Watching

Tony Fortunato  
Sr Network Performance Specialist  
The Technology Firm